

PORTFOLIO



윤해진(Harry)

Yoon Haejin

웹 프론트엔드 개발자



윤해진(Harry)

Haejin Yoon



사용자의 목소리를 귀 기울여 듣고,
그들의 필요를 반영하여 실용적인
아이디어를 코드로 구현하는 창의적인
프론트엔드 개발자 윤해진입니다.

개인정보

생년월일. 1992. 07. 18.

E-mail. hjoocomp@gmail.com

Mobile. +82 10 7262 3679

학교/전공. 청주대학교/컴퓨터공학과 졸업

성균관대학교 일반대학원/전자전기컴퓨터공학부 휴학

Programing skill

HTML/CSS



Javascript



Vue.js/Nuxt.js



Node.js



Docker



MariaDB



Python



Shiny



<https://github.com/HaejinYoon>



- 스마티소프트 (2022.03. ~ 2024.08.) (2년 6개월)



스마티소프트(주)

- JavaScript, Vue.js

- 웹 기반 사용자 정의 문서 인식 프로그램 개발



- 정보처리기사 (2017.05.26.)

- Data Analysis Technology Assessment 1급 (Python) (2025.09)

- 빅데이터분석기사 필기(2025.09.26~.)

소개(Introduction)

소개

[#Diligence]

성실함과 근면함은 저의 삶의 중요한 원칙입니다. 저는 항상 주어진 일에 최선을 다하고, 작은 일이라도 소홀히 여기지 않습니다. 매일 꾸준히 노력하며, 목표를 향해 한 걸음씩 나아가는 것을 소중하게 생각합니다.

저는 성실하게 맡은 바 임무를 수행하며, 매 순간 최선을 다하는 것을 약속합니다. 이러한 자세는 저에게 지속적인 동기부여가 되고, 나아가 나의 삶을 더욱 풍요롭게 만드는 원동력이 됩니다.

[#Punctuality]

시간 약속을 잘 지키는 것은 저의 중요한 가치 중 하나입니다. 항상 정해진 시간에 맞춰 약속을 이행하려고 노력합니다. 이를 통해 상대방에게 신뢰를 주고, 서로의 시간을 존중하는 모습을 보여주고자 합니다.



주요활동(Activities)

주요활동

LS 빅데이터스쿨 5기

- LS 미래원 (2025.07. ~ 2025.11.) (4개월)



- Python기반 빅데이터 분석
- 전기/전자/제조 도메인 지식을 반영한 데이터 분석 프로젝트 진행
- LS그룹 현직자 특강 및 멘토링

디지털 데이터 융합 JAVA 응용 SW 개발자 전문과정

- 중앙정보처리학원 (2021.09. ~ 2022.01.) (5개월)

- 국비지원 SW 교육
- 교육 마무리 프로젝트 1등 수상
- GitHub : <https://github.com/HaejinYoon/Town-Story>
- 발표자료 : <https://drive.google.com/file/d/1XmmnAVuprD7BeXnjmagfj1ami8lgMb31/view?usp=sharing>

제 2025 - LS 빅데이터스쿨 - 41호

수료증

과정명 : LS 빅데이터스쿨 5기
기간 : 2025.07.07 ~ 2025.11.05
성명 : 출해진

위 사람은 (주)LS의 LS 빅데이터스쿨 5기 과정을
성실히 수료하였기에 이 수료증을 수여합니다.

2025.11.05

LS미래원장 엄기성

총 헤더

동네 살펴보기

제작자

로그인 회원가입



Welcome to 을asticsearch!
이웃과 함께 만들어가는 스토리

▼ 더 많은 정보를 아래에서 확인하세요! ▼

주요활동

해외 자매대학 연계 어학연수(영어)

- California State University of Sacramento (2015.01. ~ 2016.01.) (1년)

- 어학당 최고 레벨 달성(Level 8).
- 본 대학 전공 관련 정규 수업 참여.



수상경력(Awards)



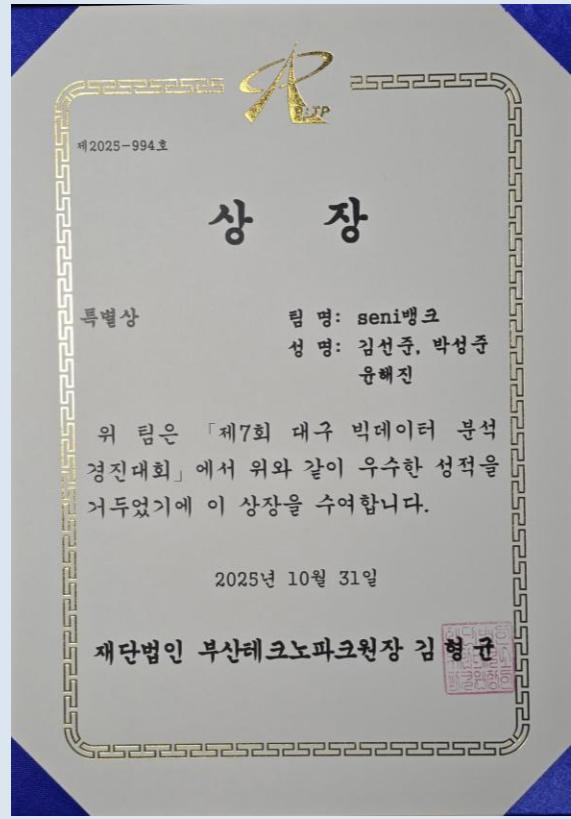
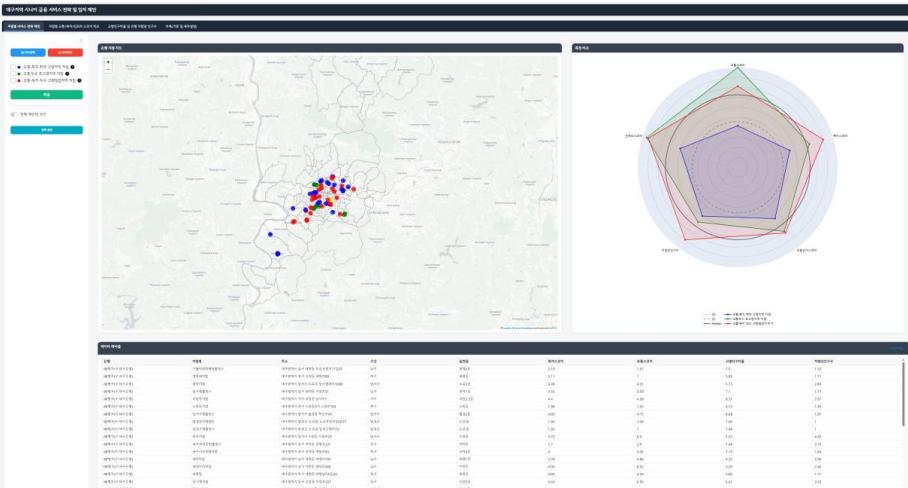
수상경력

제 7회 대구 빅데이터 분석 경진대회 : 데이터 분석 및 활용 분야 - 공공 부문

- 데이콘(DACON) (2025.08. ~ 2025.10.) (3개월)
- 특별상 수상



- 분석주제 : 대구광역시 시니어 금융 서비스 전략 및 입지 제안
 - 결과보고서 : <https://drive.google.com/file/d/1a4f0JL8fDRXnDXz65bdghdXuDdjxQUj/view?usp=sharing>
 - 발표자료 : <https://drive.google.com/file/d/1Y-IHPIdvIsD5j2UkzawPVYhSgqX5Vga/view?usp=sharing>
 - 대시보드 : https://spicywinter.shinyapps.io/pre_project2_team4



최종순위	팀명	수상	상금
1	초록대구	최우수상(대구시장상)	300만원
2	shadoway	우수상(대구시장상)	200만원
3	뛰뛰행빵	장려상(대구디지털혁신진흥원장상)	100만원
4	애플살결	입상(대구디지털혁신진흥원장상)	-
5	seni뱅크	특별상(부산테크노파크원장상)	50만원
6	클린오락실	특별상(삼성카드이사장상)	300만원
7	도다다쿵	입상(대구디지털혁신진흥원장상)	-
8	DATTO	입상(대구디지털혁신진흥원장상)	-

핵심역량(Core Competencies)

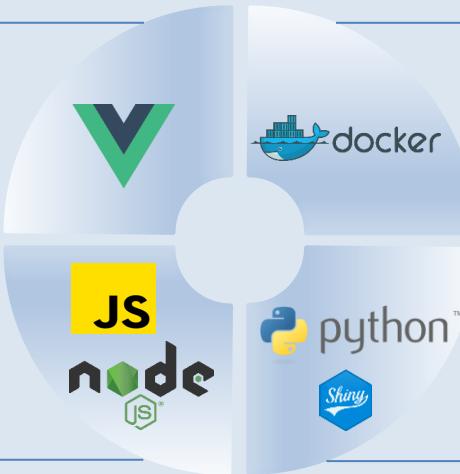
핵심역량

Vue.js

Vue.js 반응형 렌더링 구조 이해 및 활용

재사용 가능한 컴포넌트 기반 UI 설계 능력 보유

RESTful API 연동 및 비동기 통신 구현 경험 다수



Docker

Linux 환경에서 Docker를 사용한 웹 어플리케이션 구동 경험

Docker 이미지 빌드 시 다양한 커스텀 옵션 제작에 대한 경험
다수

Docker의 구동 방식에 대한 깊은 이해도

JavaScript
Node.js

UI/UX 부분 파트장으로 사용자에게 필요한 UI/UX 구성에 대한 경험

사용자와 지속적인 소통으로 요구사항을 적극 수용 및 구현

Node.js 기반 프런트, 백엔드 개발 경험

Python
Shiny

Python 기반 데이터 전처리·EDA·모델링 전반의 실무형 분석 역량 보유

Shiny·Streamlit을 통한 데이터 인사이트 시각화 및 서비스형 대시보드
구현 가능

모델 결과 분석, 시각화 UX 개선 등 데이터 해석 능력을 활용한 문제 해결
역량 보유

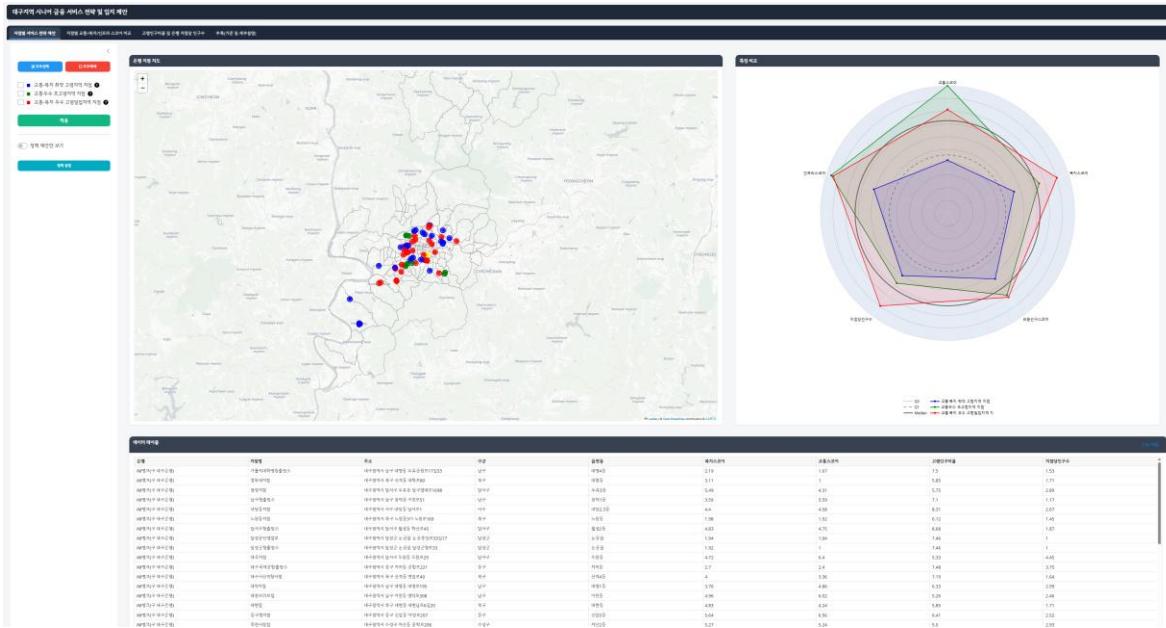
프로젝트(Projects)

프로젝트(LS 빅데이터 스쿨 기간)

:공모전 수상 프로젝트

프로젝트명	참고
1. 지역 공공데이터 분석	Github: https://github.com/HaejinYoon/project2_team4 Dashboard: https://spicywinter.shinyapps.io/pre_project2_team4/
2. 주조 공정 데이터 기반 모델 학습하기	Github: https://github.com/HaejinYoon/Project-1-Data-Driven-Modeling-of-the-Casting-Process Dashboard: https://spicywinter.shinyapps.io/main_proj1_team2/
3. 주조 공정 실시간 데이터 관리 대시보드	Github: https://github.com/HaejinYoon/LSBS_Main_Project_2_Die_Casting_Realtime_Monitoring Dashboard: https://spicywinter.shinyapps.io/main_proj_2_monitoring/
4. 전력사용에 따른 전기요금 예측 대회 및 분석 보고서 생성 대시보드	Github: https://github.com/HaejinYoon/LS_final_project Dashboard: https://electricityfareprediction.streamlit.app/

1. 지역 공공데이터 분석



주제

- #### - 대구지역 시니어 금융 서비스 전략 및 입지 제안

개요

- '시니어 특화 은행 서비스'에 초점을 맞춰, 대구시 내 기존 은행 지점들의 입지 특성 변수 도출
 - 도출된 변수들에 따라 기존 은행들 군집화 및 타겟 군집 선정
 - 타겟 군집들에 대한 입지 특징 도출 및 특징에 따른 '시니어 금융 서비스 전략' 설정
 - 각 타겟 군집 별 설정 전략에 따른 입지 제안 *입지 제안은 신규 위치 제안이 아닌 기존 지점을 분석에 따른 벤치마킹 지점 도출 방식

주요 분석 기법

- 'K-means 군집화: 기존 은행 지점들의 입지 특성을 기반으로 군집화하여 유사한 특성을 가진 지점을 그룹화
 - 커널 밀도 추정: 기존 은행 지점을 생활권 내 노인 인프라 시설 밀집도를 추정하여 스코어링
 - 위치 좌표 기반 매핑: 각 개체 요소들 위치 정보를 기반으로 지도 상에 시각화 및 분석
 - 그외 : 정성적 분석 및 기사 자료 참고

2. 주조 공정 데이터 기반 모델 학습하기

주조 공정 불량 예측

개요 데이터 탐색 예측 시뮬레이션 모델 학습

데이터 탐색



주조 공정에 대한 이해를 돋고
사용 데이터에 대한 탐색 기능을 제공합니다.

[바로가기]

예측 시뮬레이션



모델 예측 기능을 통해 입력된 공정 조건을 바탕으로
생산종의 품질 결과 예측을 제공합니다.

[바로가기]

모델 학습



개별 시도한 모델 정보와
최종 선정한 모델에 대한 설명과 근거를 제공합니다.

[바로가기]

주제

- 주조 공정 센서 데이터 기반 공정 합격·불합격 모델 개발

목적

- 주조 공정 데이터(다이캐스팅)를 활용하여 품질 불량을 예측하는 머신러닝 모델 개발
- 학습된 모델을 활용하여 사용자 입력값에 대한 실시간 예측 기능과 주요 변수 영향도 시각화 기능을 갖춘 모니터링 대시보드 구현
- 제조 현장의 품질 문제 해결을 위해 설명 가능한 인공지능 기법(XAI)을 적용하고, 실시간 적용 가능성 검증

기대효과

- 데이터 기반 객관적인 품질 관리 체계 확립
- 불량 조기 감지 및 불량률 저감 → 생산성 향상
- 설명 가능한 AI를 통한 원인 분석·관리자 의사결정 지원
- 실시간 대시보드를 통한 즉각적 대응 가능

주조 공정 불량 예측 개요 데이터 탐색 예측 시뮬레이션 모델 학습 예측 개선 방안 생산계획 입력 변수 생산 환경 정보 생산 라인: 전자교반 3라인 2호기 장비 이름: TM Carrier RH 금형 이름: TM Carrier RH-Semi-Solid DIE-06 공정 상태 관련 | | | | | |-------------|-------------|-------------|-------------| | 일조 누적 제품 개수 | 월간 누적 제품 개수 | 전체 누적 제품 개수 | 상해 구역 속도 비율 | | 148 | 13541 | 2211572 | 0 | | | | | | |-------------|----------|----------|-----------| | 주조 압력 속도 비율 | 장비 가동 여부 | 비상 정지 여부 | 측정 딜레이 여부 | | 2 | 가동 | ON | 없음 | 주, 야간 조 Day 용융 단계 용융 온도: 850°C, 용해로: A 충전 단계 주입 관 온도: 445°C, 전자 교반(EMS) 가동 시간: 19분, 하위 구간 주입 속도: 109°C/min, 상위 구간 주입 속도: 112°C/min 주방관 길이: 173m, 주입 압력: 323bar, 금형 코드: 8412 생각 단계 상부1 금형 온도: 160°C, 상부2 금형 온도: 161°C, 상부3 금형 온도: 1287°C, 하부1 금형 온도: 199°C 하부2 금형 온도: 194°C, 하부3 금형 온도: 1439°C, 네각수 온도: 32°C 공정 속도 관련 장비 전체 사이클 시간: 121s, 실제 생산 사이클 시간: 122s 예측 실행 예상 불량률: 45.96% 최종 판정: 불량제품

핵심기능

- 사용자가 직접 변수를 조정하여 설정한 조건에 따라 주조 공정을 수행할 경우, 양품과 불량 여부를 예측

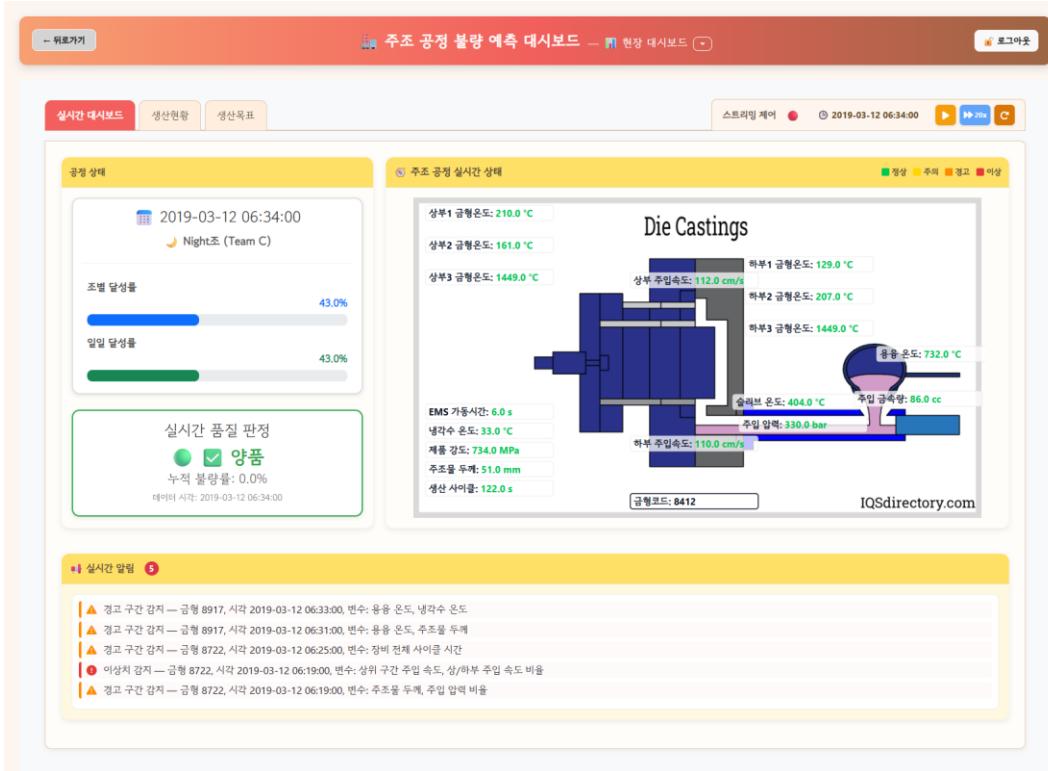
비지도 학습 기반 이상 탐지

- 정상/이상 공정 경계 정의 (PCA, DBSCAN, IsolationForest 등)
- 이상치 점수 시각화 및 최근 공정 변수 변화 감시

지도 학습 기반 불량 예측

- 분류 모델(XGBoost, LightGBM, RandomForest 등)로 불량 여부 예측
- SMOTE 등 불균형 처리 및 F1-score 기반 성능 향상
- SHAP, Permutation Importance, PDP 기반 원인 분석

3. 주조 공정 실시간 데이터 관리 대시보드



주제

- 주조 공정 실시간 품질 모니터링 및 불량 예측 대시보드

개요

- 2번 프로젝트의 연장 선상으로 주조(Die Casting) 공정의 실시간 데이터를 모니터링하고, 품질 이상 여부를 예측하여 불량을 조기에 감지하기 위한 시스템
- 센서 스트리밍 데이터를 기반으로 주·야간 조별 생산 달성을, 불량 경향, 공정별 주요 변수 모니터링, 품질 예측 및 개선 분석

핵심기능

- 사용자가 직접 변수를 조정하여 설정한 조건에 따라 주조 공정을 수행할 경우, 양품과 불량 여부를 예측

3. 주조 공정 실시간 데이터 관리 대시보드



주제

- 주조 공정 실시간 품질 모니터링 및 불량 예측 대시보드

개요

- 2번 프로젝트의 연장 선상으로 주조(Die Casting) 공정의 실시간 데이터를 모니터링하고, 품질 이상 여부를 예측하여 불량을 조기에 감지하기 위한 시스템
- 센서 스트리밍 데이터를 기반으로 주·야간 조별 생산 달성을, 불량 경향, 공정별 주요 변수 모니터링, 품질 예측 및 개선 분석

3. 주조 공정 실시간 데이터 관리 대시보드



주제

- 주조 공정 실시간 품질 모니터링 및 불량 예측 대시보드

개요

- 2번 프로젝트의 연장 선상으로 주조(Die Casting) 공정의 실시간 데이터를 모니터링하고, 품질 이상 여부를 예측하여 불량을 조기에 감지하기 위한 시스템
- 센서 스트리밍 데이터를 기반으로 주·야간 조별 생산 달성을, 불량 경향, 공정별 주요 변수 모니터링, 품질 예측 및 개선 분석

4. 전력사용에 따른 전기요금 예측 대회 및 분석 보고서 생성 대시보드



주제

- 전력 사용량 및 역률, 작업유형 등의 피처를 기반으로 전기요금 예측

개요

- 전력 사용량 데이터를 활용하여 전기요금을 예측
- 전력 사용 추세 분석, 피크 수요 진단, 자동 보고서 생성 기능이 포함된 Streamlit 대시보드

4. 전력사용에 따른 전기요금 예측 대회 및 분석 보고서 생성 대시보드

The screenshot displays the LS ELECTRIC real-time power monitoring system. It includes a sidebar for 'Real-time Power Monitoring System' with a 'Production Speed' section showing values of 19.00, 0.25, and 20.00. Buttons for '▶ 시작' (Start) and '초기화' (Reset) are present. A red dot indicates '정지' (Stop). The main interface features the LS ELECTRIC logo and navigation tabs: HOME, 실시간 모니터링, 통계 분석 (selected), and 부록. Date range selection is set from 2024/11/25 to 2024/12/01. A search bar and a 'Report Generation' button are also visible. The '주요 지표' (Key Indicators) section shows four boxes: 전력사용량 (17,768.2 kWh, -10.0%), 전기요금 (2,362,849 원, -4.0%), 평균 단가 (133.0 원/kWh, +17.4%), and 탄소배출량 (5.88 tCO₂, -10.0%). Below this is a note about the comparison period: 비교 주간: 전월 동일 기간 2024-10-25 ~ 2024-11-01. The '보고서 생성' (Report Generation) button is highlighted in red. The '요일-시간대별 전력사용 패턴' (Daily-Hourly Power Consumption Pattern) section contains two charts: a bar chart for average daily power consumption by day of the week (Monday to Sunday) and a line chart for average hourly power consumption over time.

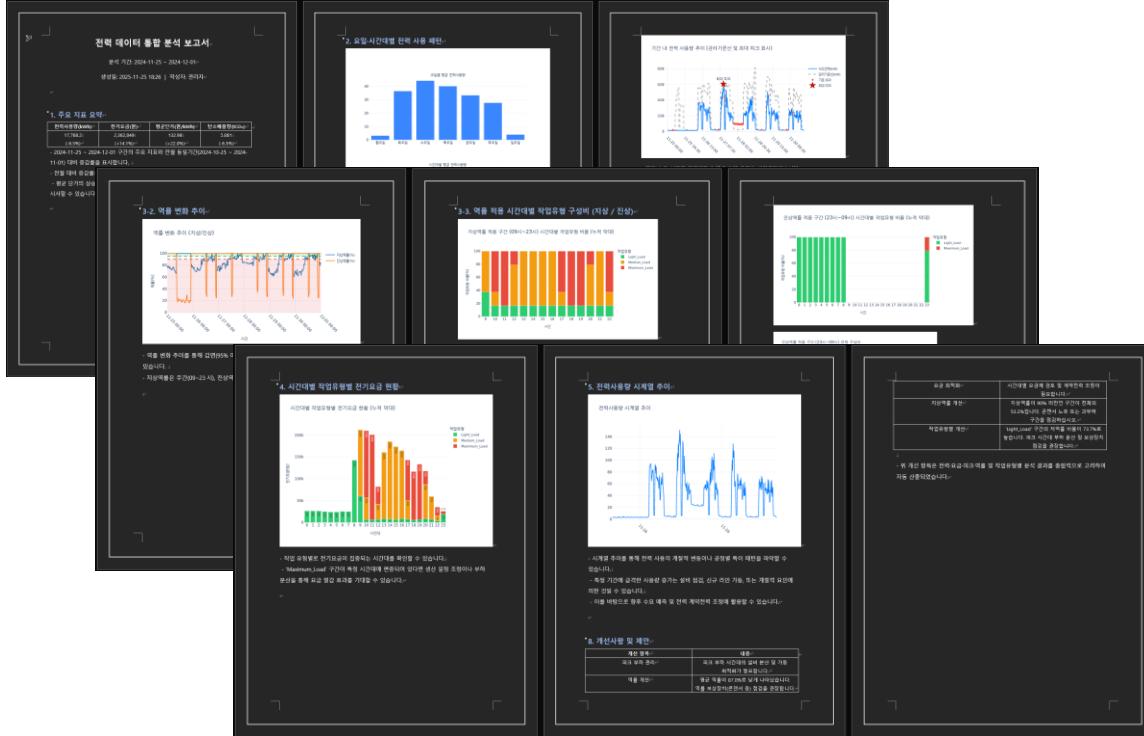
요일	평균 전력사용량 (kWh)
월요일	~5
화요일	~35
수요일	~45
목요일	~40
금요일	~30
토요일	~25
일요일	~5

시간대	평균 전력사용량 (kWh)
00:00 ~ 05:00	~10
05:00 ~ 10:00	~10
10:00 ~ 15:00	~65
15:00 ~ 20:00	~55
20:00 ~ 25:00	~45
25:00 ~ 00:00	~10

통계분석

- 기간별 전력 사용 추이
- 요일·시간대별 평균 사용량 Heatmap
- 피크 수요 표시 마커
- 단가 변화 추이 Dual Line Chart
- 역률 비교 Gauge / Scatter

4. 전력사용에 따른 전기요금 예측 대회 및 분석 보고서 생성 대시보드



보고서 생성 기능

모듈: report_generator.py

형식: .docx (Word 문서)

포함 내용:

- 주요 지표 요약
- 요일/시간대별 전력 사용 패턴
- 피크 수요 지표
- 전력 사용량 추이 그래프
- 개선 제안 자동 생성

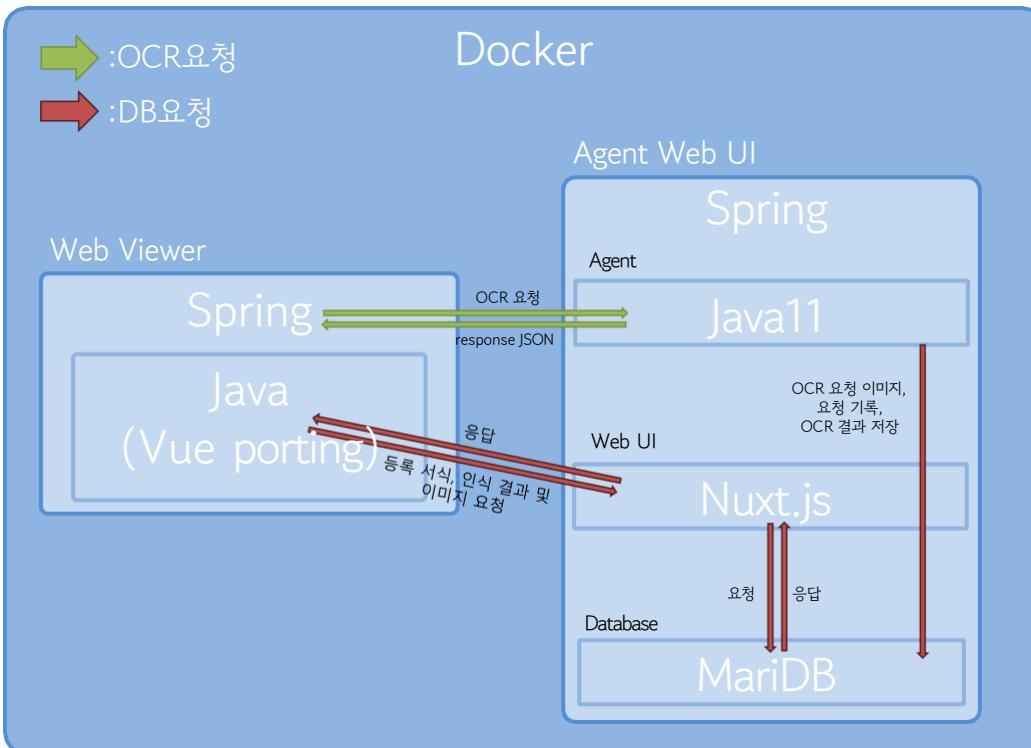
프로젝트(경력사항)

프로젝트명	기간	역할	적용 기술	설명
1. SAHARA Web Viewer	2024.01. ~ 2024.08.	UI/UX 파트장 프런트, 백엔드 통합 개발	JavaScript(Node.js), Vue.js, Express.js, Docker, MariaDB, HTML, CSS	Web Viewer 백, 프런트 엔드 통합
2. Agent Web UI	2022.06. ~ 2024.08.	제품 관련 유지 보수	JavaScript(Node.js), Nuxt.js, Express.js, Docker, MariaDB, Tomcat Spring, HTML, CSS	OCR 통계, 관리 프로그램
3. ARES-SAHARA Web Viewer	2022.03. ~ 2023.12.	프런트 엔드 개발	JavaScript(Node.js), Vue.js, Docker, Tomcat Spring, HTML, CSS	웹 브라우저 기반 사용자 정의 문서 인식 어플리케이션

:주력 프로젝트

1. SAHARA Web Viewer

기존 구조



Project

- 기존 문서 인식 어플리케이션의 프런트, 백엔드 통합
프로젝트 (SAHARA Web Viewer)

개발 인원 및 기간

- 개발 기간 : 2024.01. ~ 2024.08.
- 개발 인원 : 3명

적용 기술

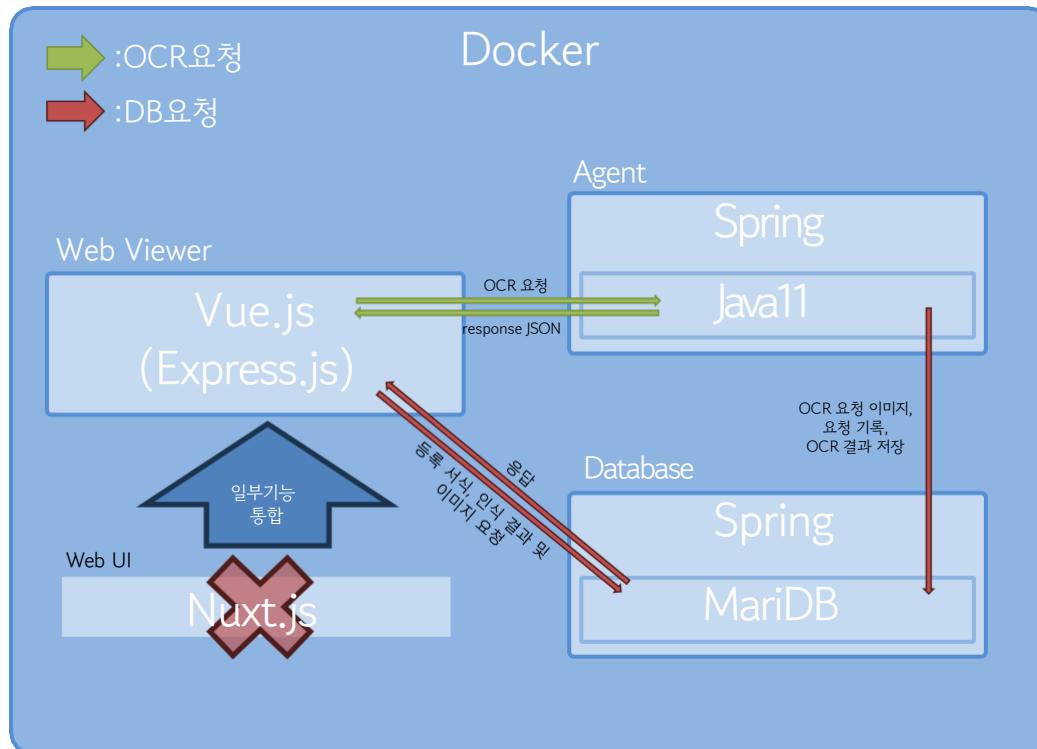
- JavaScript(Node.js)
- Vue.js
- Express.js
- Docker
- MariaDB
- HTML
- CSS

설명

- 기존 ARES-SAHARA Web Viewer 프로젝트에서 큰 역할이 없던 JAVA로 이루어진 백엔드를 제거하고 실제로 백엔드로 사용하던 Agent Web UI 프로젝트의 백엔드 파트를 분리하여 하나의 단일 프로젝트로 통합.
- 정형 문서 제외, 비 정형 문서 학습을 위한 데이터 셋 업로드 및 학습 기능 추가.

1. SAHARA Web Viewer

개선 구조

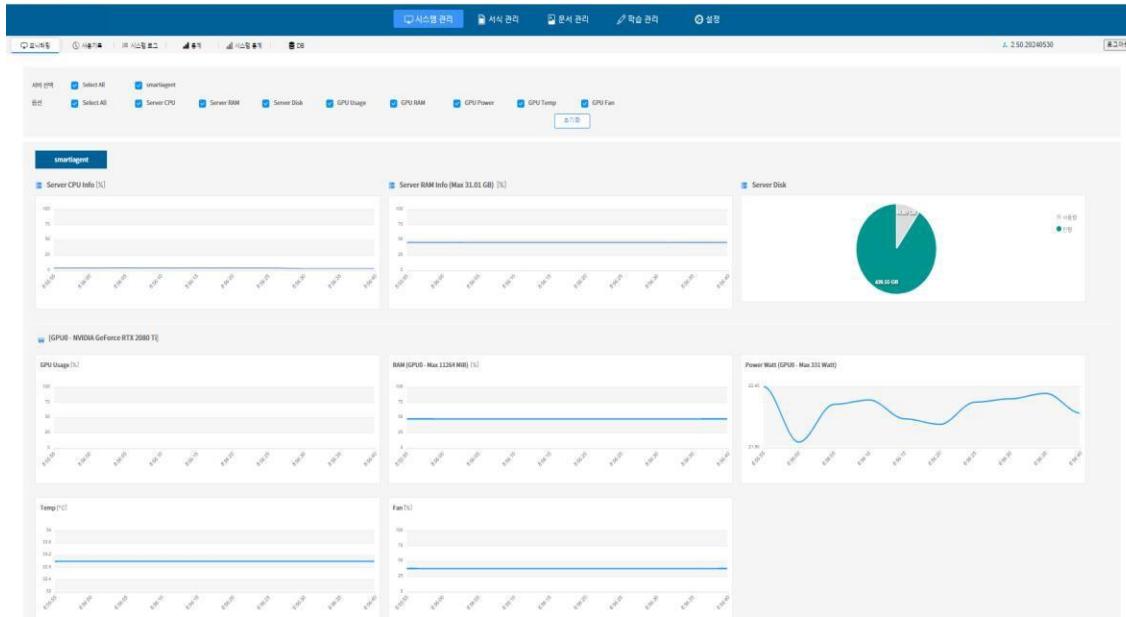


주요 성과

- Web Viewer의 프런트, 백엔드를 JS로 통일하면서 개발 편의성 향상.
- 여러 구조를 합쳐 놓아 용량이 너무 방대했던 이미지 가 세분화 되며 이미지 빌딩 시간 단축.

1. SAHARA Web Viewer

서버 리소스 모니터링



화면 개요

- 서버 리소스 모니터링 화면

화면 기능

- 현재 웹 어플리케이션이 구동하는 시스템에 대한 정보 (CPU 점유율, GPU 점유율, 하드디스크 용량 등)를 제공하는 UI 구성.

1. SAHARA Web Viewer

라벨링 화면



화면 개요

- 딥러닝 학습을 위한 라벨링 화면

화면 기능

1. 키(Q)

- 키 버튼 클릭 또는 단축키(Q)를 눌러
- 활성화 된 상태에서 OCR 블록을 클릭

2. 벨류(W)

- 벨류 버튼 클릭 또는 단축키(Q)를 눌러
- 활성화 된 상태에서 OCR 블록을 클릭

3. 삭제(Del)

- 키 또는 벨류가 선택된 상태에서 삭제 버튼
- 클릭 또는 단축키 Del 클릭

4. OCR 이미지 위 라벨링 키, 벨류 표시

5. 라벨링 이름 입력

6. 키 벨류 값 수정 및 태그 설정

7. 라벨링 저장 버튼

8. 라벨링 완료 버튼

1. SAHARA Web Viewer

라벨링 학습 화면

The screenshot shows the SAHARA Web Viewer interface with the '학습 관리' (Learning Management) tab selected. The main area displays two learning progress charts: one for 'f1 score' and another for 'loss'. The left sidebar includes sections for '상태 정보' (Status Information), '데이터 관리' (Data Management), and 'NER별 학습 설정' (NER-specific Learning Settings). The 'NER별 학습 설정' section contains various configuration parameters like learning rate, batch size, and epoch number.

화면 개요

- 사용자가 등록한 라벨링 또는 데이터셋 학습을 위한 화면

화면 기능

- 라벨링으로 등록한 키, 벨류로 학습을 진행.
- 또는 JSON 형식으로 이루어진 키, 벨류 데이터셋을 업로드, 전송하고 학습 진행 현황을 확인할 수 있는 UI 구성.

1. SAHARA Web Viewer

체크박스 학습 화면

1 상태 정보

API 버전 1.0
.ml 적용중인 딥러닝 모델 버전 default_model
GPU 상태 GPU 1: 11264.0MB

2 학습 목록

데이터셋 이름	전체 파일수	True 파일수	False 파일수	상성율
default	68422	34211	34211	2024.06.14 00:04
demo_data	3313	1204	2109	2024.06.14 00:04

3 학습할 데이터셋 demo_data

4 딥러닝 모델 목록

모델 이름	최근학습 일자	최근학습시 정확도	최근학습시 검증정확도	학습한 데이터셋
default_model	2024.03.14 10:34	100.00%	0.98%	default
default_model2	2024.04.08 16:49	92.00%	0.89%	default
test	2024.06.20 00:09	88.65%	89.65%	demo_data, demo_data, demo_data, demo_data, demo_data,

5 학습할 딥러닝 모델

모델 이름	최근 학습 일자	최근 학습시 정확도	최근 학습시 검증정확도	학습한 데이터셋
default_model	2024.03.14 10:34	100.00%	0.98%	default

학습 이미지 다운로드(zip) 적용 삭제

화면 개요

- 등록된 서식에서 체크박스 데이터를 학습 데이터로 등록하고 학습을 진행

화면 기능

1. 상태 정보

- API버전 정보 표시
- 적용중인 딥러닝 모델 버전 정보 표시
- GPU 상태 표시

2. 데이터셋 목록

- 데이터 목록 표시
- 새로운 데이터 업로드
- 데이터 삭제

3. 학습할 데이터셋

- 학습 목록에서 선택한 데이터를 표시

4. 딥러닝 모델 목록

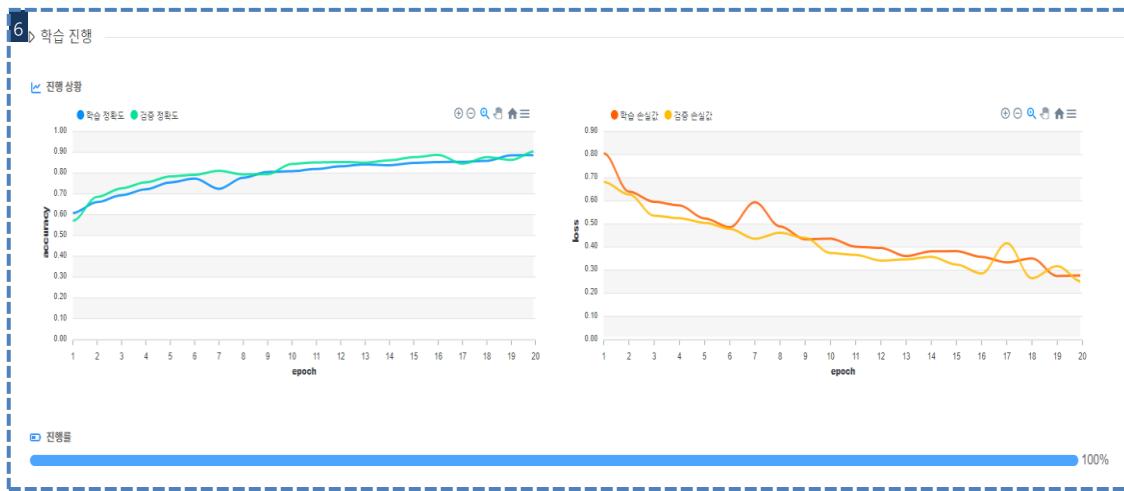
- 딥러닝 모델 목록 표시
- 체크박스 이미지 다운로드
- 딥러닝 모델 목록 적용
- 딥러닝 모델 삭제

5. 학습할 딥러닝 모델

- 딥러닝 모델 목록에서 적용한 모델 표시

1. SAHARA Web Viewer

체크박스 학습 화면



화면 개요

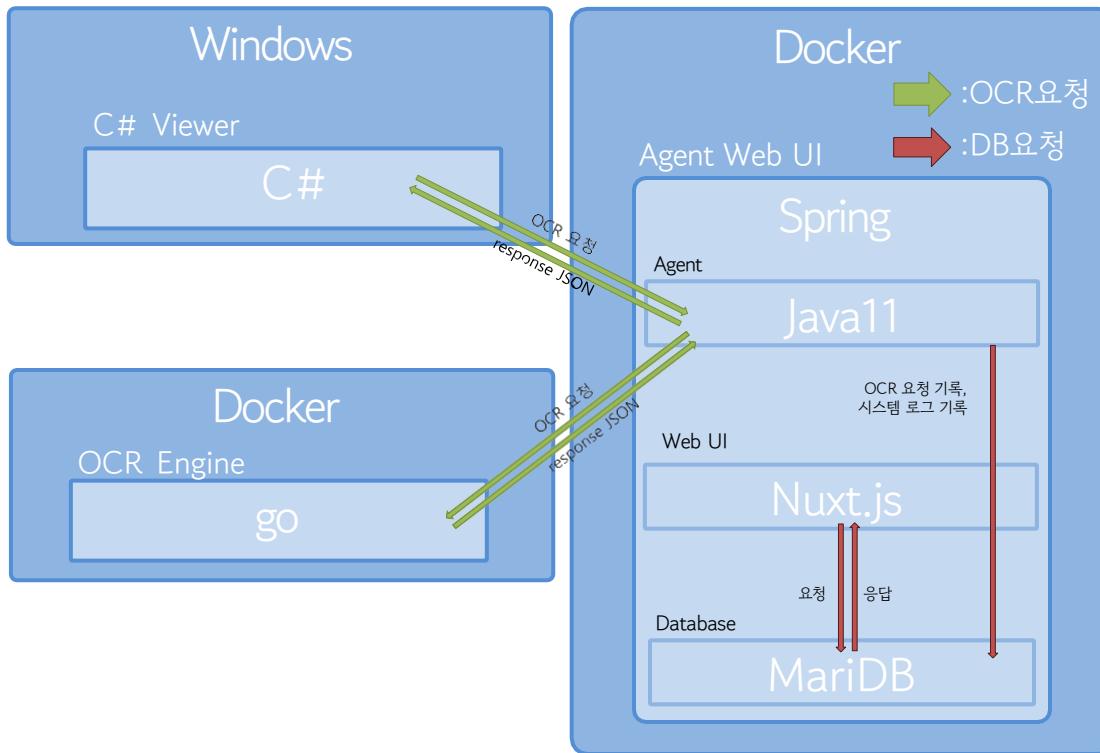
- 등록된 서식에서 체크박스 데이터를 학습 데이터로 등록하고 학습을 진행

화면 기능

6. 학습 진행
- 체크박스 학습 진행 상태를 그래프로 표시

2. Agent Web UI

어플리케이션 구조



Project

- OCR 관리 어플리케이션 (Agent Web UI)

개발 인원 및 기간

- 개발 기간 : 2022.06. ~ 2024.08.
- 개발 인원 : 2 명

적용 기술

- | | |
|-----------------------|-----------------|
| - JavaScript(Node.js) | - MariaDB |
| - Nuxt.js | - Tomcat Spring |
| - Express.js | - HTML |
| - Docker | - CSS |

설명

- 제품 정기 점검 및 고객사에서 이슈 발생 시 점검 및 업데이트
- 제품 업데이트 시 고객사 방문하여 최신 버전으로 업데이트

2. Agent Web UI

통계 화면 최적화

The screenshot shows the Agent Web UI interface. At the top, there is a navigation bar with tabs: '홈화면' (Home), '사용기록' (Usage Record), '시스템 흐름' (System Flow), '로그' (Log), '시스템 초기화' (System Initialization), and '도움' (Help). The date '2024.05.30' and a '로그인' (Login) button are also present.

Below the navigation bar, there are several filter sections:

- 기준:** '페이지별' (Page-based) and '모정별' (Customer-based) radio buttons.
- 서식 기준:** '서식' (Format) and '세부서식' (Detailed Format) radio buttons.
- 서식 검색:** A text input field with placeholder '서식 이름을 입력해주세요.' (Enter format name).
- 사용자 이름:** A text input field with placeholder '사용자 이름을 입력해주세요.' (Enter user name).

Under '서비스 선택' (Service Selection):

- 'Select All' checkbox
- '모든 서비스(보기)' (View all services) checkbox

Under '옵션' (Options):

- 'Select All' checkbox
- '사전대입 사용방' (Pre-filled usage method) checkbox
- '일별 사용방' (Daily usage method) checkbox
- '주별 사용방' (Weekly usage method) checkbox
- '월별 사용방' (Monthly usage method) checkbox
- '연식 성공률' (Annual success rate) checkbox
- '부서별 사용방' (Departmental usage method) checkbox
- '서식 인식률' (Format recognition rate) checkbox

Under '종류' (Type):

- 'All' checkbox
- 'OCR' checkbox
- 'ODR' checkbox

At the bottom left, there is a '기간선택' (Date Selection) section with two date pickers: '2024-05-18' and '오전 12:00', and a time picker '00:00 ~ 01:27'. Below these are two buttons: '선택' (Select) and '다음' (Next).

The main content area displays a grid of charts:

- 정밀 사용방**: 2024-05-18, 정밀 사용방(페이지별, 모정 서비스)
- 일별 사용방**: 2024-05-18, 일별 사용방(페이지별, 모정 서비스)
- 주별 사용방**: 2024-05-18, 주별 사용방(페이지별, 모정 서비스)
- 일별 사용방**: 2024-05-18, 일별 사용방(페이지별, 모정 서비스)
- 서부별 사용방**: 2024-05-18, 서부별 사용방(페이지별, 모정 서비스)
- 부서별 사용방**: 2024-05-18, 부서별 사용방(페이지별, 모정 서비스)
- 연식 성공률**: 2024-05-18, 연식 성공률(페이지별, 모정 서비스)

Each chart provides visual data representation for the selected service and period.

주요 성과

- 고객사 시스템에서 진입 조차 불가능 했던 통계 화면을 최적화 하여 원활한 사용이 가능하도록 개선.
- 고객사에서 자사의 프로그램에 대한 긍정적인 인상을 심어주어 추후 제품 수주에 영향

2. Agent Web UI

통계 화면 최적화

```
3450 case '/api/history-subfield':  
3451   if(date.checkedServers.indexOf('0') < 0){  
3452     break  
3453   }  
3454   let arr = []  
3455   for (let i = 0; i < date.checkedItems.length; i++) {  
3456     for (let j = 0; j < date.subfieldActiveSwitchList.length; j++) {  
3457       if(date.checkedItems[i]==0 || date.checkedItems[i]==4){ // 서브필드에는 시간대별 사용량이 없기 때문에 스킵  
3458         break;  
3459       }  
3460       let field = date.subfieldActiveSwitchList[j].  
3461       server_0 = await history.getGroupBySubfield(  
3462         date.startDate,  
3463         date.endDate,  
3464         date.date[date.checkedItems[i]-1],  
3465         field,  
3466         date.tag  
3467       )  
3468       arr.push(server_0)  
3469     }  
3470   }  
3471   infoData.push(arr)  
3472   // console.log('/api/history-subfield>>> ", JSON.stringify(infoData))  
3473   break
```

```
237   getGroupBySubfield(start, end, date, field, tag) {  
238     return new Promise((resolve) => {  
239       let selectQuery = ""  
240       let subfield = ""  
241       field = parseInt(field)  
242       switch (field) {  
243         case 1: subfield = "sub_field_1"; break;  
244         case 2: subfield = "sub_field_2"; break;  
245         case 3: subfield = "sub_field_3"; break;  
246         case 4: subfield = "sub_field_4"; break;  
247         case 5: subfield = "sub_field_5"; break;  
248       }  
249       if (date == "day") {  
250         selectQuery = `count(*) as cnt, ` + subfield + ` as name, DATE_FORMAT(create_dt,'%Y-%m-%d') as date`  
251       } else if (date == "week") {  
252         selectQuery = `count(*) as cnt, ` + subfield + ` as name, CONCAT(DATE_FORMAT(create_dt,'%Y-%m'),'-',  
253       ) else if (date == "month") {  
254         selectQuery = `count(*) as cnt, ` + subfield + ` as name, DATE_FORMAT(create_dt,'%Y-%m') as date`  
255       }  
256     })  
257   }  
258 }  
259  
260 module.exports = {  
261   getGroupBySubfield  
262 }  
263  
264 // module.exports = require('./agentWebUI')
```

문제 원인

- 데이터 집계 로직 및 UI의 문제
- 통계 화면 진입 시 모든 통계 데이터를 만들고 UI상에서 체크한 항목만 보여주는 방식
- 방대한 데이터량으로 인해 충분한 시스템 사양이 반쳐주지 못하면 UI 로딩이 불가

해결 방법

- 데이터 집계 로직 변경
- 통계 화면 진입 시 기본값(1주일)의 데이터량만 로딩 후 필요에 따라 검색 조건(field)을 변경한 뒤 검색을 할 경우 새로 데이터를 집계

2. Agent Web UI

신규 SSO 로그인 대응

```
245 ✓ mounted(){
246 ✓   if(this.$route.query.ticket != null){
247 ✓     this.submitSso();
248 ✓   } else if(this.$route.query.empid != null && this.$route.query.ssid == sessionStorage.getItem("secureSessionId")){
249 ✓     this.loginProc();
250 ✓   }
251 ✓   if(this.$route.query.withoutSso == 'true'){
252 ✓     this.withoutSso = true
253 ✓   }
254 }
```

```
163 ✓ submitSso(){ // data에 주후 ipAddress: await this.getAddress()파트 추가
164   // let data = {
165   //   ticket: escape(this.$route.query.ticket),
166   // }
167   this.loading = true
168   let data = {
169     ticket: unescape(this.$route.query.ticket),
170   }
171
172 ✓ createLoginSso(data).then((res) => {
173   // console.log('createLoginSso :: res', res)
174   this.loading = false
175   if(res.status === 200) {
176     if(res.data.status == 409) {
177       return alert(res.data.message)
178     }
179   let data = {
180     accesstoken: res.data.infoData,
181   }
182   //일정 기간마다 패스워드 변경, 기간 변경은 routes/api.js의 router.post('/account')
183   if(res.data.changePassword){
184     this.$store.dispatch('store/loginChangePw', data)
185   }else{
186     this.$store.dispatch('store/login', data)
187   }
188   // window.alert(window.location.host.split(":")[0]);
189   } else {
190     let message = res.data.message
191     alert(message)
192   }
193 }).catch((err) => {
194   this.loading = false
195   console.log(err)
196 })
197 // console.log('this.$route.query.ticket >>', this.$route.query.ticket)
198 },
```

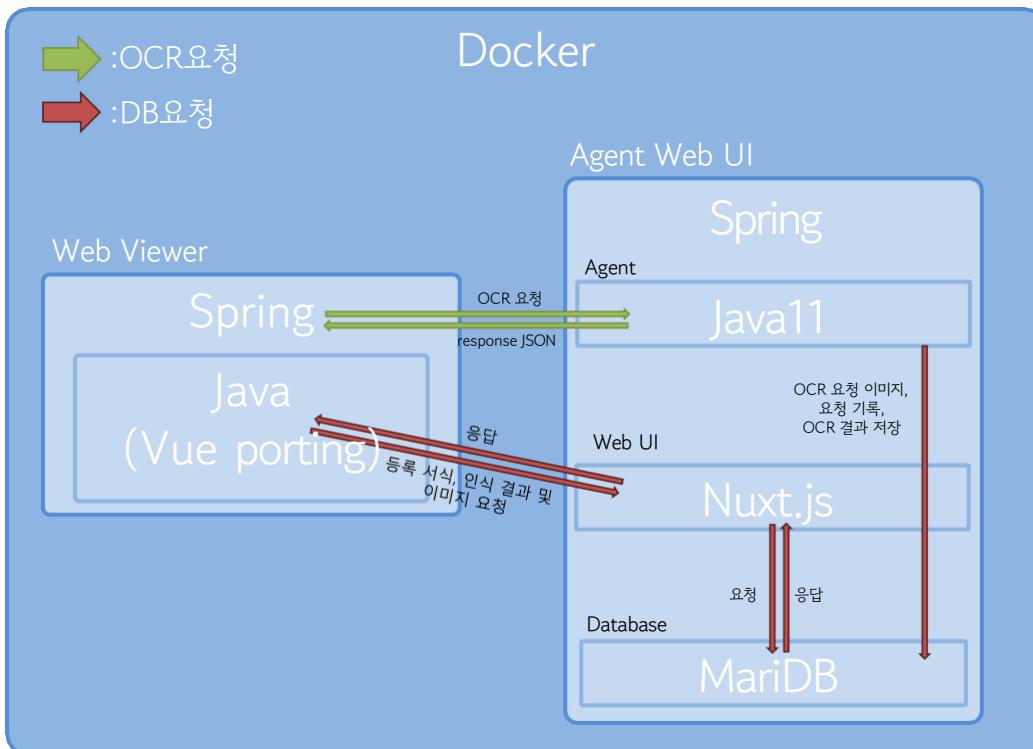
```
285 loginProc(){
286   try {
287     let empInfo ={
288       empid: this.$route.query.empid
289     }
290     this.loading = true
291     console.log("loginProc started. empinfo >> ", empInfo)
292     console.log("loginProc started. type of empinfo >> ", typeof(empInfo))
293     createLoginMwSso(empInfo).then((res) => {
294       if(res.status == 200) {
295         if(res.data.status == 409) {
296           return alert(res.data.message)
297         }
298         let data = {
299           accesstoken: res.data.infoData,
300         }
301         //일정 기간마다 패스워드 변경, 기간 변경은 routes/api.js의 router.post('/accounts/login')에서
302         if(res.data.changePassword){
303           this.$store.dispatch('store/loginChangePw', data)
304         }else{
305           this.$store.dispatch('store/login', data)
306         }
307         // window.alert(window.location.host.split(":")[0]);
308         } else {
309           let message = res.data.message
310           alert(message)
311         }
312       }).catch((err) => {
313         this.loading = false
314         console.log(err)
315       })
316     }catch(err){
317       console.log('loginProc error', err)
318     }
319 }
```

주요 성과

- 고객사의 SSO 시스템 변경으로 인한 대응
- 구식 html 방식에서 JS 방식으로 변경
- 고객사에서 각 SSO인증 과정에서 요청, 응답 값을 제공하지 않아 대응에 어려움이 있었음
- 요청, 응답 값을 예측하여 코드를 작성하고 테스트는 고객사 내부에서 시행한 후 다시 수정해야 했던 개발 과정이 매우 번거로웠던 작업

3. ARES-SAHARA Web Viewer

어플리케이션 구조



Project

- 웹 브라우저 기반 사용자 정의 문서 인식 어플리케이션 (ARES-SAHARA Web Viewer)

개발 인원 및 기간

- 개발 기간: 2022.03. ~ 2023.12.
- 개발 인원: 4 명

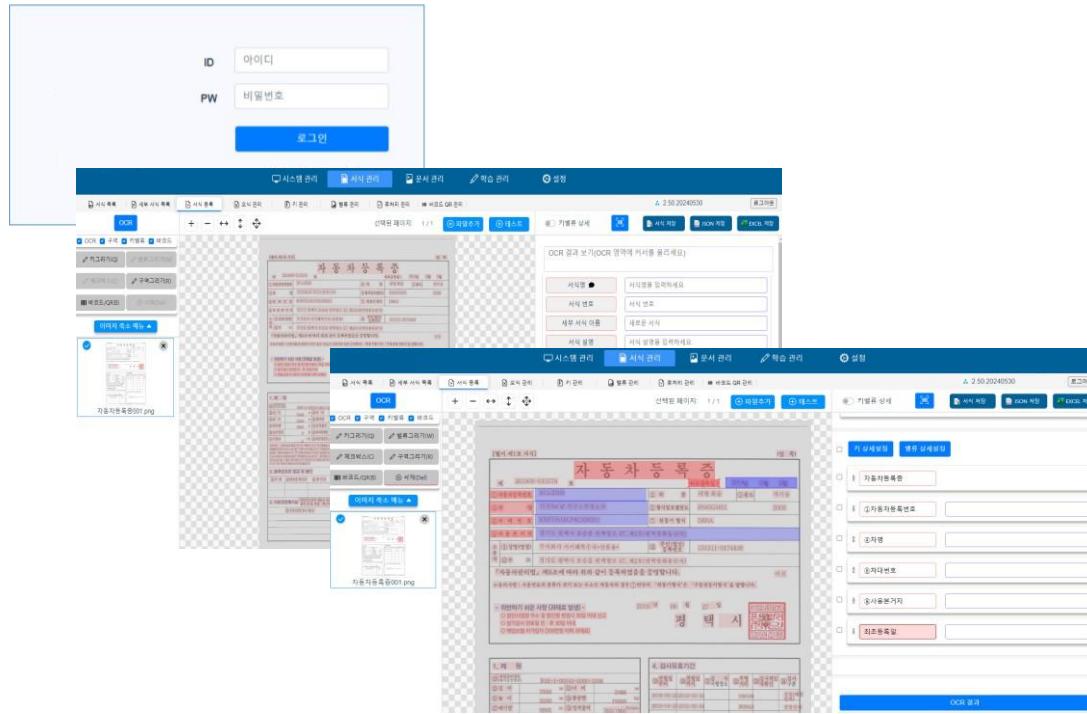
적용 기술

- | | |
|-----------------------|-----------------|
| - JavaScript(Node.js) | - Tomcat Spring |
| - Vue.js | - HTML |
| - Docker | - CSS |
| - MariaDB | |

설명

- 사용자 정의 문서 인식 어플리케이션 제작.
- 사용자가 지정한 Key, Value의 위치를 통하여 문서를 인식하고 분류.

3. ARES-SAHLARA Web Viewer



주요 성과

- 계정을 통하여 허가된 사람만 사용할 수 있도록 계정 인증 환경 구성
 - 사용자가 문서를 정의하는데 필요한 화면 UI 구성.
 - 사용자 정의 문서를 수정하고 관리하는 화면 UI 구성.
 - 문서 인식 결과를 확인하는 화면 UI 구성.
 - 문서 인식 결과를 DB화하여 상태별로 이를 관리하고 편집할 수 있는 UI 구성 및 제작.
 - 문서 인식 요청 주소, 2번 프로젝트의 Agent 원격 업데이트 등 서버 설정 페이지 제작

3. ARES-SAHLARA Web Viewer

The screenshot displays the ARES-SAHLARA Web Viewer interface. At the top, there is a navigation bar with tabs: '시스템 관리' (System Management), '서식 관리' (Form Management), '문서 관리' (Document Management) (which is selected), and '학습 관리' (Learning Management). Below the navigation bar, there is a search bar and a date indicator '2024.05.09 12:49'. On the left, there is a sidebar with sections for 'ODR 요청 목록' (ODR Request List) and 'ODR 요청 (0)' (0 ODR Requests). The main content area shows a large button to 'Add file' and a message '파일을 추가하려면 클릭 또는 드래그 하세요.' (Click or drag to add file). Below this, there is a section for '요청 파일 리스트' (Request File List) with counts for '인식 완료' (0), '미등록 문서' (0), and '인식 실패' (0). The central part of the screen shows a table titled '최근 요청 기록 (20)' (Recent Request Log (20)) with columns: '처리 태입', '서식 이름', '세부 서식 이름', '페이지', '처리 상태', '이미지', '처리 일시', and '관리'. A red dotted line highlights the '처리 일시' column. A red box highlights the '관리' column for the first row. A modal window titled '오류 보고' (Error Report) is open at the bottom, containing fields for '이미지' (Image) with a preview of '사업자등록증001.png', '서식 종류' (Form Type) with '사업자등록증', and '설서 내용 작성' (Note Content) with '오류 내용을 상세히 작성해주세요.' (Please provide detailed error content). Buttons for '취소' (Cancel) and '저장' (Save) are at the bottom of the modal.

주요 성과

- 계정을 통하여 허가된 사람만 사용할 수 있도록 계정 인증 환경 구성
- 사용자가 문서를 정의하는데 필요한 화면 UI 구성.
- 사용자 정의 문서를 수정하고 관리하는 화면 UI 구성.
- 문서 인식 결과를 확인하는 화면 UI 구성.
- 문서 인식 결과를 DB화하여 상태별로 이를 관리하고 편집할 수 있는 UI 구성 및 제작.
- 문서 인식 요청 주소, 2번 프로젝트의 Agent 원격 업데이트 등 서버 설정 페이지 제작

3. ARES-SAHLARA Web Viewer

The screenshot displays the ARES-SAHLARA Web Viewer interface. At the top, there are tabs for System Management, Document Management, and Learning Management. The main area shows a document titled "사업자등록증명" (Business Registration Certificate) with various fields like name, address, and registration number. To the right, an "OCR 결과 표시" (OCR Result Display) panel shows extracted text from the document, such as "사업자등록증명01" and "사업자등록증명02". Below this are sections for saving and viewing the results. On the left, there's a sidebar for managing files and a preview of the scanned document.

주요 성과

- 계정을 통하여 허가된 사람만 사용할 수 있도록 계정 인증 환경 구성
- 사용자가 문서를 정의하는데 필요한 화면 UI 구성.
- 사용자 정의 문서를 수정하고 관리하는 화면 UI 구성.
- 문서 인식 결과를 확인하는 화면 UI 구성.
- 문서 인식 결과를 DB화하여 상태별로 이를 관리하고 편집할 수 있는 UI 구성 및 제작.
- 문서 인식 요청 주소, 2번 프로젝트의 Agent 원격 업데이트 등 서버 설정 페이지 제작

3. ARES-SAHLARA Web Viewer

The screenshot displays two pages of the ARES-SAHLARA Web Viewer interface.

Top Navigation Bar: Includes links for System Management, Document Management, and Settings, along with a timestamp (2.50.20240530) and a Logout button.

Service Configuration Page: Shows a service named "SmartAgent" with the following details:

- Server Details:** ID: smartisoft, License: 67a89184 - ecfcc6491, Name: SmartAgent, URL: http://smartagent, Port: 8080, Type: Agent.
- Buttons:** Add (+), Delete (-), Import (Import), Export (Export), and Save (Save).

Document Management Page: Shows a section for "Document Upload" with a file input field and a "Browse" button.

주요 성과

- 계정을 통하여 허가된 사람만 사용할 수 있도록 계정 인증 환경 구성.
- 사용자가 문서를 정의하는데 필요한 화면 UI 구성.
- 사용자 정의 문서를 수정하고 관리하는 화면 UI 구성.
- 문서 인식 결과를 확인하는 화면 UI 구성.
- 문서 인식 결과를 DB화하여 상태별로 이를 관리하고 편집할 수 있는 UI 구성 및 제작.
- 문서 인식 요청 주소, 2번 프로젝트의 Agent 원격 업데이트 등 서버 설정 페이지 제작.

3. ARES-SAHLARA Web Viewer

서식 등록 화면



화면 개요

- 사용자가 정의하는 문서 등록 화면

화면 기능

1. 그리기 선택 툴

- 클릭 또는 단축키를 통하여 원하는 형식(키, 밸류 등)의 사각형 그리기

2. 메인 캔버스

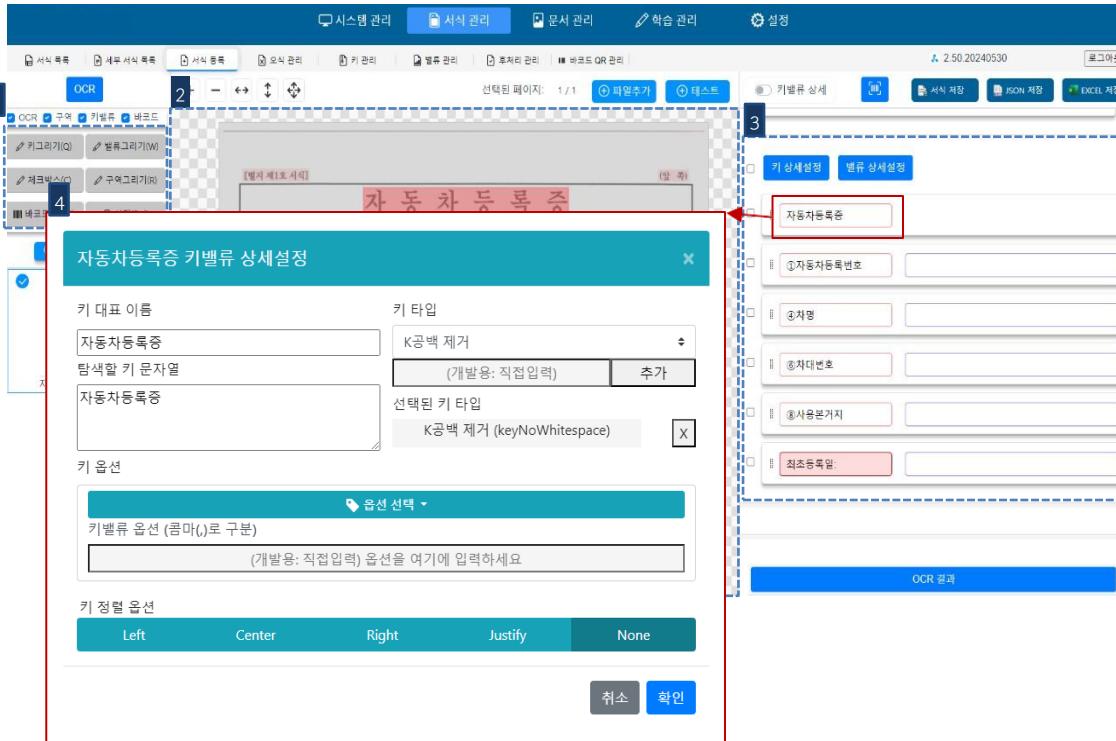
- OCR이 진행된 이미지가 그려지는 캔버스
 - 1.의 툴을 사용하여 영역을 지정하면 사각형 내부의 문자가
자동으로 3.의 리스트 영역에 입력

3. 키, 밸류, 체크박스, 구역 리스트

- 사용자가 추가한 키, 밸류, 체크박스, 구역의 리스트가 나타나며 각 항목을 클릭하면 세부사항 설정 가능

3. ARES-SAHLARA Web Viewer

서식 등록 화면



화면 개요

- 사용자가 정의하는 문서 등록 화면

화면 기능

1. 그리기 선택 툴

- 클릭 또는 단축키를 통하여 원하는 형식(키, 밸류 등)의 사각형 그리기

2. 메인 캔버스

- OCR이 진행된 이미지가 그려지는 캔버스
- 1.의 툴을 사용하여 영역을 지정하면 사각형 내부의 문자가 자동으로 3.의 리스트 영역에 입력

3. 키, 밸류, 체크박스, 구역 리스트

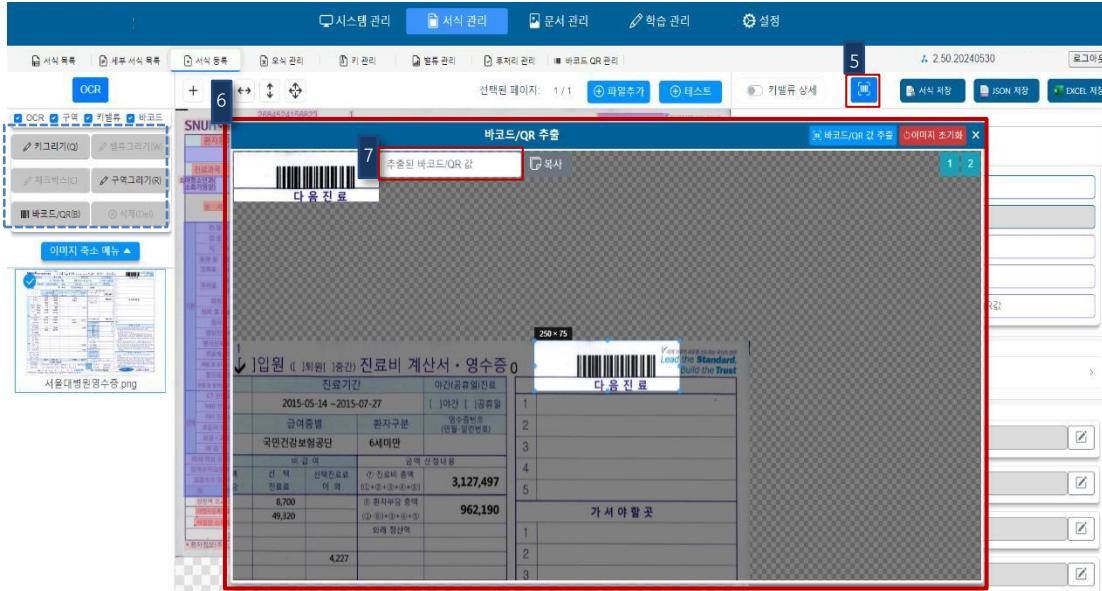
- 사용자가 추가한 키, 밸류, 체크박스, 구역의 리스트가 나타나며 각 항목을 클릭하면 세부사항 설정 가능

4. 키에 관련된 세부 사항 설정

- 대표 이름, 탐색할 키 문자열, 타입 등 세부 사항을 설정하는 팝업
- 밸류, 체크박스, 구역도 이와 유사한 설정 화면 보유

3. ARES-SAHLARA Web Viewer

서식 등록 화면



화면 개요

- 사용자가 정의하는 문서 등록 화면

화면 기능

5. 바코드/QR 추출 팝업 버튼

- 서식의 바코드/QR 추출 화면을 불러오는 버튼

6. 바코드/QR 추출 화면

- 이미지의 바코드 또는 QR 구역을 크롭하여 추출 버튼을 클릭하면 Agent에서 그 값을 추출하여 7.에 표시

3. ARES-SAHLARA Web Viewer

문서 관리 화면

화면 개요

- 문서 인식을 요청하고 인식 요청 결과를 확인하는 화면

화면 기능

1. 인식된 결과 상태 별 확인

- 각 상태의 탭을 클릭하여 해당 상태에 대한 결과를 확인 가능

2. 파일 추가 구역

- 인식을 진행할 파일을 드래그 앤 드롭 또는 클릭으로 탐색기를 열어 직접 지정하여 추가

3. 문서 인식 진행 현황

- 사용자가 추가한 파일들에 대한 인식 진행 현황을 나타내는 리스트

4. 최근 요청 기록

- 최근 요청한 20개의 기록을 바로 확인 가능한 리스트

5. 오류 보고 버튼

- 해당 결과에 대한 오류 보고를 시행 할 수 있는 버튼.

3. ARES-SAHLARA Web Viewer

결과 편집 화면



화면 개요

- 인식된 결과를 편집하고 상태 변경을 수행하는 결과 편집 화면

화면 기능

1. 이미지 조정 툴 박스

- 이미지의 확대, 축소, 폭 맞춤, 높이 맞춤, 회전, 원상 복구

2. 메인 캔버스

- 이미지 인식 결과를 그림으로 표현

3. 결과 표시 창

- 서식명, 결과 처리 상태, 인식 성공한 키, 밸류를 표시해주는 결과 창
- 인식 등급에 따라 색상 표시
- 결과 행의 추가 또는 삭제

4. 돋보기 뷰

- 해당 결과가 이미지에서 어느 부분에 해당하는지 확대 표시
- 마우스 휠로 확대/축소

5. 결과 영역 편집 툴

- 인식된 결과의 영역이 의도한 바와 다를 때 영역 편집을 통하여 정확한 영역을 편집/저장

3. ARES-SAHARA Web Viewer

결과 편집 화면



화면 개요

- 인식된 결과를 편집하고 상태 변경을 수행하는 결과 편집화면

화면 기능

6. 결과 편집 버튼

- 결과의 저장과 동시에 결과의 상태(인식완료-수정완료-컨펌완료-최종컨펌완료 순)변경
 - 삭제 시 휴지통 상태로 변경, 휴지통 리스트에서 완전 삭제
 - 오류 보고 버튼

7. 기타 메뉴

- JSON 저장 : 현재 결과를 JSON 형태로 로컬 저장
 - 색상 설정 : 캔버스에 표시되는 결과의 색상 설정
 - EXCEL 저장 : 현재 결과를 EXCEL로 로컬 저장
 - 체크박스 편집 : 체크박스 편집 툴 팝업 여부
 - 체크박스 지정 : 현재 이미지에서 체크박스 학습에 사용될



Thank You



윤 해 진(Harry)
hjyoomp@gmail.com